GB04/4619

GB/04/4619

INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

PRIORITY DOCUMENT

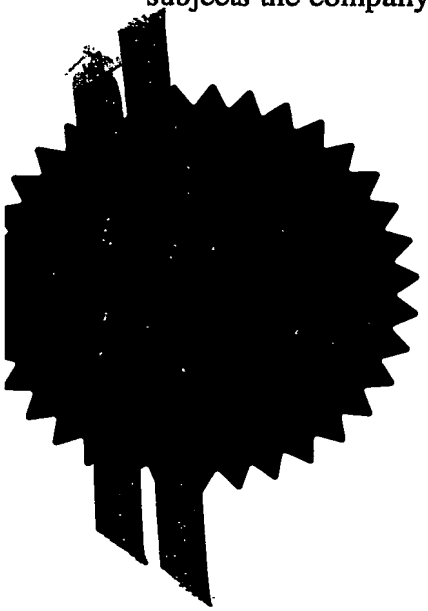
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.



Signed R. Mahoney

Dated 10 November 2004

For Official use only

03NOV03 E848818-1 D10092
P01/7700 0.00-0325504.9

THE PATENT OFFICE
J
31 OCT 2003
LONDON

31 OCT 2003

Your reference **Security Eng (UK)**

0325504.9

The **Patent Office**

Request for grant of a Patent

Form 1/77

Patents Act 1977

1 Title of invention

Security Engineering: A process for developing accurate and reliable security systems

2. Applicant's details

[X] First or only applicant

2a If applying as a corporate body: Corporate Name

Country

2b If applying as an individual or partnership

Surname **Leach**

Forenames **John**

2c Address

Innisfree
Stoke Road
Smannell
Andover
Hants

UK Postcode SP11 6JL

Country GB

ADP Number 8744922001

☐	Second applicant (if any)	
2d	Corporate Name	
	Country	
2e	Surname	
	Forenames	
2f	Address	
	UK Postcode	
	Country	
	ADP Number	
3	Address for service	
	Agent's Name	Origin Limited
	Agent's Address	52 Muswell Hill Road London
	Agent's postcode	N10 3JR
	Agent's ADP Number	C03274

7270457002

4 Reference Number

Security Eng (UK)

5 Claiming an earlier application date

An earlier filing date is claimed:

Yes ☐ No ☒

Number of earlier application or patent number

Filing date

15 (4) (Divisional)	8(3)	12(6)	37(4)
☐	☐	☐	☐

6 Declaration of priority

Country of filing	Priority Application Number	Filing Date

7 Inventorship

The applicant(s) are the sole inventors/joint inventors

Yes [X] No [ ]

8 Checklist

		Continuation sheets	
Claims	0	Description	50
Abstract	0	Drawings	0

Priority Documents	~~Yes~~/No
Translations of Priority Documents	~~Yes~~/No
Patents Form 7/77	~~Yes~~/No
Patents Form 9/77	~~Yes~~/No
Patents Form 10/77	~~Yes~~/No

9 Request

We request the grant of a patent on the basis of this application

Signed: Origin Limited Date: 30 October 2003

(Origin Limited)

Security Engineering

A Process for Developing Accurate and Reliable Security Systems

31st October 2003

Prepared By:
Dr John Leach
☎ = 01264 332 477
✆ = 07734 311 567
✉ = John.Leach@JohnLeachIS.com

Contents

This paper has been written to support the patent application being made by Dr John Leach for the protection of the intellectual property contained herein. All the new intellectual property contained herein is the exclusive proprietary property of Dr John Leach.

1. SUMMARY

This paper describes a process provisionally named "Security Engineering" which has been created by Dr John Leach. Security Engineering is an improved process for the quantification and modelling of Information Security threats and their interactions with security countermeasures.

Security Engineering is of value to organisations wishing to improve the way they manage their Information Security risk management arrangements. It can be used for the development of security systems which the security designer can demonstrate satisfy specified measurable security targets. This will enable security systems to be designed with increased reliability and accuracy, and will enable business management to measure the benefits to their business of their Information Security expenditures.

Security Engineering is also of value for those companies that wish to provide Information Security products and services to the marketplace. It will enable them to provide more effective products and better focused services. This includes security product vendors, Digital Risk insurance providers, providers of commodity security assurance services, security management services, threat intelligence services, and so forth.

This paper describes the Security Engineering process and the steps and stages it comprises. The central innovative idea at the heart of Security Engineering is the idea of describing threats according to a number density function and describing security countermeasures according to how they interact with and modify the threat number density. Countermeasures are classified into a small number of types and their interactions with the threat number density described in terms of resistance, mitigation and alleviation functions, each of which is specifically defined within the Security Engineering technique. Statistical techniques are used to model the threat-countermeasure interactions, including stochastic techniques for situations where a successful attack reinforces or strengthens the subsequent threat in the vicinity of the attack.

Security Engineering models the interactions between threats and security measures in a way that enables the security designer to derive the likelihood, severity and other characteristics of the resultant security incidents as a function of the capabilities and strength of the applied security measures. Security measures are described in a form that enables their effectiveness to be quantified. This enables the properties of the resultant security incidents, including their likelihood, to be calculated as a function of the degree to

which different security measures are applied. This forecasting ability enables the benefit of each security measure to be calculated and enables business management to make decisions regarding the degree to which security measures might be deployed on the basis of quantified and reliable analysis. Information Security expenditure decisions can be supported by the analysis, senior management can meaningfully benchmark their security programmes against business-driven metrics, and company boards can demonstrate to stakeholders that they have an appropriate and cost-efficient security programme in place as part of their fulfillment of their growing Corporate Governance obligations.

2. INTRODUCTION

A. THE PROBLEM THAT SECURITY ENGINEERING ADDRESSES

Information Security practitioners have long sought for methods that will enable them to show in a quantifiable and measurable way the effects of applying security solutions, controls or measures (hereinafter collectively called security measures or countermeasures) on the level and nature of information security incidents experienced.

At the present time, practitioners do not have satisfactory ways to:

- Quantify the different degrees or extents to which security measures can be applied;
- Forecast how the nature, likelihood or other characteristics of security incidents would be affected by the application of a given degree of any security measure;
- Quantify the security risk or other measure of insecurity in a business-driven manner in a form against which a security design can be measured objectively.

As a consequence of these shortcomings, senior business management does not have adequate methods for:

- Directing security designers using objective and measurable expressions for the targets a required security design should satisfy;
- Evaluating with confidence and with provability whether a given security programme will provide adequate protection to the business, or of measuring the outcomes attributable to a given security programme;
- Evaluating the benefits to the business or the return on investment (RoI) achieved by a given expenditure on a security programme;
- Demonstrating to stakeholders (including shareholders and regulators) that the senior management of the company has an appropriate and cost-efficient security programme in place given the security needs of the business.

B. CURRENT METHODS FOR ADDRESSING THIS PROBLEM

There are various methods which practitioners use when trying to analyse and assess risk, none of which convincingly provides objective measures of risk. Risk has proven to be a very difficult construct to work with, mainly because security practitioners do not have an adequate understanding of how to analyse security incidents when most types of security incident appear to be:

- Essentially unique;
- Very unlikely actually to occur;
- Rarely fully captured and described.

There is no standard or accepted method for quantifying security measures. Practitioners might be able generally to agree what constitutes more or less of any given security measure, but there is no accepted yardstick or measurement scheme for quantifying the amount or degree of security measure implemented. Evaluation criteria (e.g. the Common Criteria) specify an ordered series of evaluation levels but these levels are not set according to a scale measuring objectively the outcomes that each level of security measure would achieve.

There is no standard or accepted method for quantifying the effect of a given degree of any particular security measure on the nature, likelihood or other characteristic of a given type of security incident. Practitioners might have a general expectation that applying "more" (in whatever form) of any particular security measure should reduce the level of relevant security incidents experienced, but there is no accepted scheme for calibrating or measuring the effect of a given level of security measure on a given type of security incident.

When designing a security scheme, practitioners generally select security measures on the basis of security policy, established current or best practice, and their personal experience. They estimate the magnitude of the particular security risk (usually through a risk assessment or impact assessment) and select, by following corporate policy or established local practice, a set of measures which they accept to be appropriate for addressing those risks. There is no mechanism for showing in a measurable way the level of protection that the selected measures provide or of demonstrating that the policy-driven or practice-driven selection process leads to an optimum or efficient set of security measures. Senior management can determine that the company's security expenditure is effective if the company does not experience an intolerable level of security problems. However, senior management would

not know if they could have achieved that outcome with a significantly reduced level of security expenditure.

Senior management makes a judgement call regarding the company's intended level of spend on security measures, usually driven by the level of comfort they wish to have that serious and externally-visible security incidents will not occur. Other than pointing to the level of incidents actually experienced, and very few organisations have sound practices in place for capturing data about the security incidents they experience, senior management, if challenged, would not have mechanisms by which it could demonstrate to stakeholders that it is spending sufficiently or efficiently with its stated security programme.

Similarly, regulators and legislators do not have adequate means to set minimum outcome-driven security standards on the companies under their jurisdiction. It is usual for regulation and legislation to express security requirements in terms of subjective and immeasurable goals, as in the use of terms such as "reasonable" or "appropriate". (See for example the Data Protection Act 1988 which obliges companies to take "appropriate technical and organisational measures" to protect information [Principle 7 of 8]). In the absence of any effective means to correlate desired security outcomes with the security measures needed to achieve those outcomes, it is left to the courts to build up through case history an interpretation of what might constitute "appropriate" security measures. There is very little case history in place at the present time.

C. THE CAPABILITIES OF SECURITY ENGINEERING

Security Engineering is a process and set of methods and techniques for creating an objective and measurable expression of the business' security need and for modelling the interactions between threats and security measures in a manner which will enable designs for security measures or security programmes to be:

- Well focused – by extracting the security design targets from a suitable analysis of the process capabilities that are important to the business and which most need protection;
- Provable – by having the security designer design to meet specific measurable security targets set by the security procurer;
- Accurate – by enabling the designer to tailor and scale the design according to the known and measured threat environments;

- Reliable – by enabling the designer to base the security design on the known effectiveness of each security component at resisting, mitigating or alleviating relevant threats, thereby reducing the risk of either under- or over-engineering the security design;

- Effective and cost-efficient – by enabling the designer to optimise the security design in order to minimise, in accordance with stated business priorities, the cost of the design or the impairment of other operational requirements such as performance, ease of use, reliability.

Calculations can be independently verified to give management assurance. The effectiveness of a security programme can be measured objectively and its success or failure at satisfying the set security targets can be established according to agreed criteria. Management can maintain an incident response capability geared to achieve an agreed service level satisfying stated response time targets against expected levels of security incident.

The Return on Investment expected or achieved by the resultant security design can be calculated by costing the design and comparing that with the forecast or measured benefits achievable or achieved as a direct result of the proposed or actual security expenditure. Security Engineering will bring improved transparency to the way that Information Security supports the business, and senior business management will be able to strengthen its oversight and supervision of its Information Security arrangements.

3. SECURITY ENGINEERING DESCRIBED

A. OVERVIEW

The Security Engineering process is described with reference to Figure 1 and Figure 2. Figure 1 shows the process in its entirety.

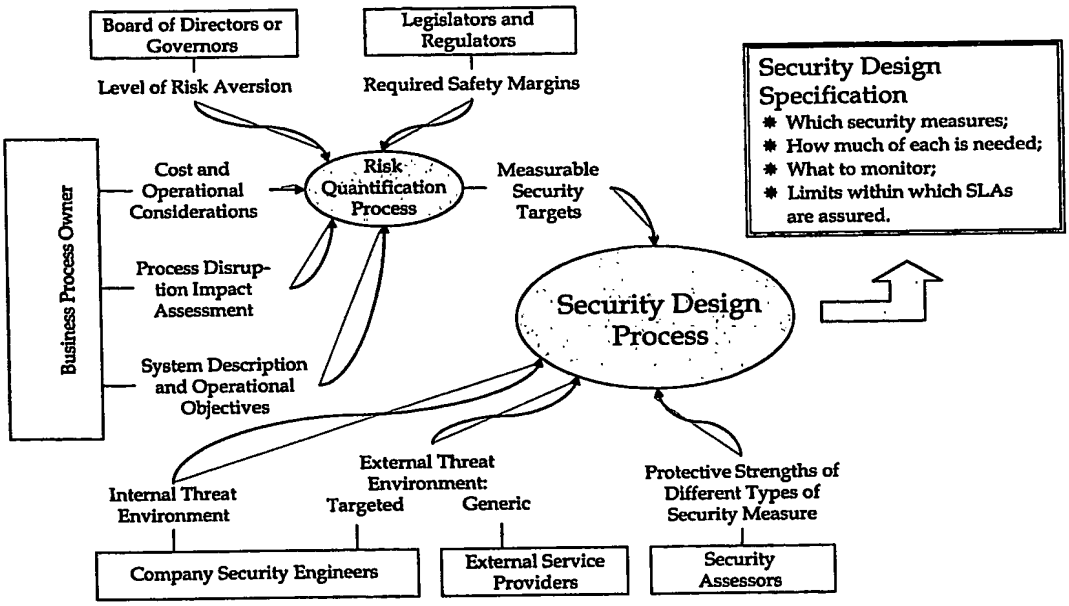


Figure 1: Security Engineering in Overview

As shown, the design of a security scheme requires contributions from a variety of actors:

- The Business Process Owner (BPO) – the representative of the business unit owning the business process being protected. The BPO is responsible for setting the security targets for the desired security scheme and for accepting or not the resultant security design;
- The board or other internal governance body - which represents the internal policy-setting authority. This body, at a macro level, determines the relative priority to be given to security compared to other information management requirements and sets the baseline for the level of business protection required;

- Legislators and regulators - who can impose minimum standards of practice for certain situations, for example for compliance with data protection legislation or the protection of retail customers;
- The corporate security engineer - who advises on the threat environments as experienced by the corporate entity (the internal threat and the targeted external threat) and provides a quantified description of the relevant threats to be used as input to the security design;
- External service providers - who would provide advice on the generic external threat environment that prevails, and provide a quantified description of the relevant threats (primarily viruses, worms and threats to the Internet infrastructure) to be used as input to the security design;
- Security assessors - who provide a quantified description of the security capabilities and operational impacts of each security measure. These assessors might be internal to the corporate entity, technical representatives of security product vendors, consultants, independent assessors, or a combination of these;
- The security designer - who models the interactions between threats and security measures to determine an optimum security design satisfying the security targets set by the BPO.

Figure 2 shows in overview the four constituent steps of the Security Engineering process, some of which comprise several stages.

Step 1 is the process of defining the measurable security targets that the security designer will aim to satisfy.

Step 2 is the process of analysing and measuring the threat environments.

Step 3 is the process of calibrating applicable security measures for their effects on the threats and other considerations.

Step 4 is the process of modelling the interactions between the threat forces and the security measures, selecting security measures in order to satisfy the security targets set, and optimising the security design.

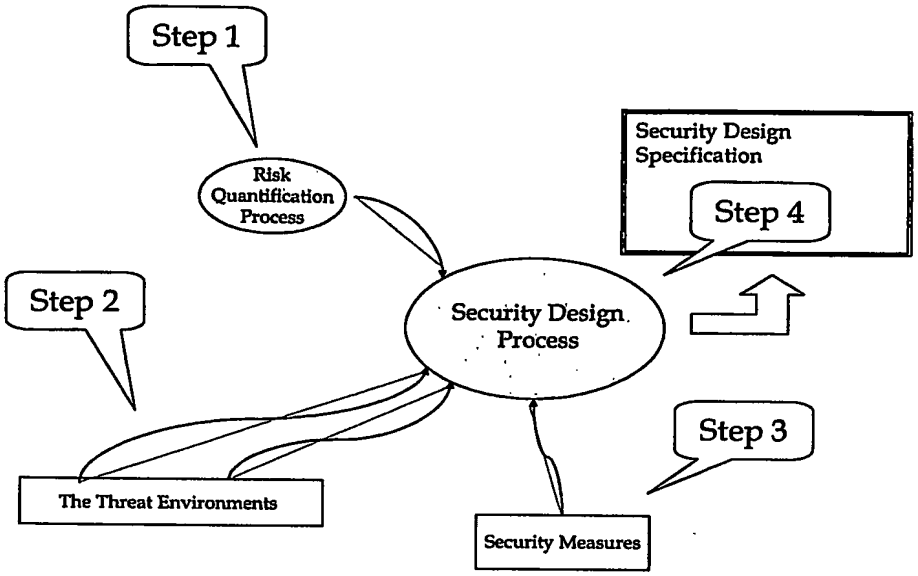


Figure 2: The Four Steps in Overview

B. THE MODELLING TECHNIQUE

Security Engineering models the interactions between threats and security measures in order to calculate the likelihood and other useful characteristics of the resultant security incidents. Threats are described in the form of a number density in several variables. Security measures are described in terms of corresponding variables. The modelling derives the expected probability of the resultant incidents and severity of the resultant process disruptions as a function of the strength of the security measures employed. The modelling uses statistical techniques, including stochastic techniques, to describe the evolution of the threat number density as the threats engage with the security measures, and is capable of modelling the evolution of the threat number density for situations where a successful attack feeds back into the magnitude of the subsequent threat, for example when modelling viruses (where a successfully infected host then becomes part of the virus threat environment for its neighbours) or internal threats (where staff are the threat agents and their inclination towards unauthorised activity is influenced by what they see of the activities of others in their working environment).

Threats are described and quantified in terms of variables which are appropriate for modelling their interactions with security measures. The selection of appropriate variables follows from the analysis of the security measures employed to protect against the threat. In describing the Security Engineering

modelling technique, we will start by classifying security measures according to type and will then describe how Security Engineering models the three types of interactions between the threat number density function and each type of security measure.

I. THE CLASSIFICATION OF SECURITY MEASURES

For the purposes of modelling, security measures are classified according to the following three classes. Most security measures fall naturally into one of the three classes; they are clearly either Interdictive, Mitigative or Alleviative measures. There are some security measures that might reach across more than one class. These can, for the purposes of modelling, be treated as if they were two separate and independent countermeasures, each of which is classified as being either Interdictive, Mitigative or Alleviative, and their interactions with the threat number density modelled separately.

The three security measure classes are as follows.

a) Interdictive measures –

These are measures which, either by deterring the threat or by resisting the threat, make successful attacks less likely. There are few opportunities to deter the external threat but there are several opportunities to deter the internal threat.

Examples of Interdictive measures include:

- Those deterring the threat:
  - User Security Education (reducing the number of people likely to cause an inadvertent security incident through error or omission)
  - Security Culture (reducing the number of people likely to commit negligence, wilful irresponsibility, or misuse of privileges)
- Those resisting the threat:
  - Anti Virus (reducing the probability of virus exposure causing infection)
  - Authentication and Access Control
  - Firewalls

- Patching
- Platform hardening

b) Mitigative measures

These measures do not make successful attacks less likely to occur. They work, instead, to make the process disruption resulting from a security incident less severe, either by working directly on the threat and reducing the intrinsic severity of the disruptions that threat is likely to cause, or by adding resilience to the protected system so the threat is not able to achieve its full potential severity.

Examples of Mitigative measures include:

- Incident Response (including vendor support agreements for the replacement of faulty equipment)
- Monitoring (detection of successful attacks)
- Redundancy
  - Of equipment
  - Of process (e.g. checks and balances, checksums, authentication codes)

c) Alleviative measures

These measures do not make successful attacks less likely to occur, or make their results less severe. They work, instead, to alleviate the impact of induced process disruptions.

Examples of Alleviative measures include:

- Fallback systems – an alternative means for achieving mission critical process functions in times when the main process is unable to complete these functions. For example, faxing out invoices when the electronic delivery mechanism is disrupted.
- Insurance

Each threat being modelled will be described by a number density which is a function of three variables plus time, where each one of the three variables is

used for modelling the interaction of the threat with one of the three types of security measure.

We shall look first at modelling the interactions which determine the success or failure of the threat at inducing a security incident. We will then discuss modelling the interactions that determine the severity of the disruption caused by the security incident, then the impact of the disruption.

II. MODELLING THE LIKELIHOOD OF SECURITY INCIDENTS

a) How Interdictive Security Measures Work

The interactions which determine the success or failure of an attack are the interactions between the threat and the Interdictive countermeasures. These are modelled as a contest between a threat with a certain capability to penetrate defences and a countermeasure with a certain capability to resist the threat.

We define α to be the variable which describes the ability of the threat to penetrate security defences and β to be the variable which describes the ability of the countermeasure to resist that penetration. The threat number density function, n, gives the number density of elements within the threat environment at each level of penetrability α. Hence, $n = n(\alpha)$ (ignoring the other variables, including time, for this part of the discussion).

A threat element generally wins (and gives rise to a security incident) when its ability to penetrate defences (the α value at which that element resides within the density function) clearly dominates[1] the countermeasure's ability to resist penetration (the β value for that instance of defence). It generally loses (no incident results) when its ability to penetrate is clearly dominated by the countermeasure's ability to resist penetration. For threat elements where the threat's penetrability is well matched by the countermeasure's resistance, the probability of a successful attack is an intermediate value between 1.0 (certain success) and 0.0 (certain failure).

We define the probability per unit time of successful penetration for a threat of strength α against an Interdictive measure of strength β as $P(\alpha,\beta)$. We define for each countermeasure a resistance function, which we denote $R(\alpha,\beta)$, where $P(\alpha,\beta) = 1 - R(\alpha,\beta)$. $P(\alpha,\beta)$ is close to 1.0 when α clearly dominates β

[1] We use the term "dominates" rather than "is greater than" for the following reason. We may well choose to define the α and β variables for the interaction in a form where a small α indicates high penetrability and a small β indicates high resistance. In such as case, to say "α clearly dominates β" means α is much less than β rather than α is much greater than β.

and is close to 0.0 when β clearly dominates α. It makes a smooth transition between the two when α and β are of similar magnitude. The transition is smooth, not a vertical transition from $P(\alpha,\beta) = 1.0$ to $P(\alpha,\beta) = 0.0$ at $\alpha = \beta$, to reflect variability in the interaction, i.e. those particular unknowable contributions which have an influence on the success of the attack at the time of the contest and which contribute to making each incident that occurs essentially unique.

If there are no interdictive measures in place, $P(\alpha,\beta) = P(\alpha) \equiv 1$, i.e. any exposure to the threat leads to a successful attack.

The likelihood per unit time of an incident occurring is given by the function

$$L(\beta) = \int n(\alpha).P(\alpha,\beta).\partial\alpha.$$

$L(\beta)$ describes how the likelihood of an incident occurring varies with the resistive strength, β, of the countermeasures applied.

b) Calculating the Interdictive Variables and Functions

For this modelling technique to have value, we need to have confidence that we will be able to determine the variables α and β and the functions $n(\alpha)$ and $P(\alpha,\beta)$ for any threat and countermeasure of interest. α, β, and $P(\alpha,\beta)$ are obtained from an analysis of the interaction between threat and security measure. For a given threat and security measure, these need be determined only once. Once they have been agreed by consensus amongst security practitioners, they are set and do not need to determined afresh for each analysis. $n(\alpha)$ describes the threat within the prevailing threat environment and will vary from threat environment to threat environment and perhaps from day to day or month to month. It is obtained by measurement.

A sketched-out example will serve to illuminate this analysis.

(1) A SIMPLE ANTI-VIRUS EXAMPLE

Consider the situation of a stand-alone desktop host protected by an Anti-Virus (AV) product being exposed to e-mail-borne viruses. The ability of the AV product to prevent any exposure to a virus (the threat) causing an infection on the desktop (an incident) is determined by the promptness with which the user updates their desktop's AV signature files. (This presumes, which is a fair approximation in today's mature AV marketplace, that an AV product will have a 100% rate of success at resisting any virus that is recorded in its signature files.)

In this case, β, the variable that describes the resistive strength of the Interdictive measure, can be set to be the number of hours between successive signature file updates. The virus threat's penetrability, α, corresponding to this selection for β, is then the age of the virus in hours since it was first released.

There is one other time variable of importance in this interaction, T_{update}, which is the typical length of time it takes, in hours, for the AV vendor to get a new virus' signature into its signature file measured from the moment of release of the virus into the threat environment. In today's marketplace, T_{update} can be set at around 20 hours, with AV vendors taking as much as 40 hours for no more than a small proportion of viruses. T_{update} is an average update delay, and for each individual virus the actual T_{update}, if it could be known precisely, would vary around that average. This variability contributes to the smoothing of $R(\alpha,\beta)$ around $\alpha = T_{update}$ and $\alpha = T_{update} + \beta$.

As a result we have $R(\alpha,\beta)$ described approximately by the curve shown in Figure 3.

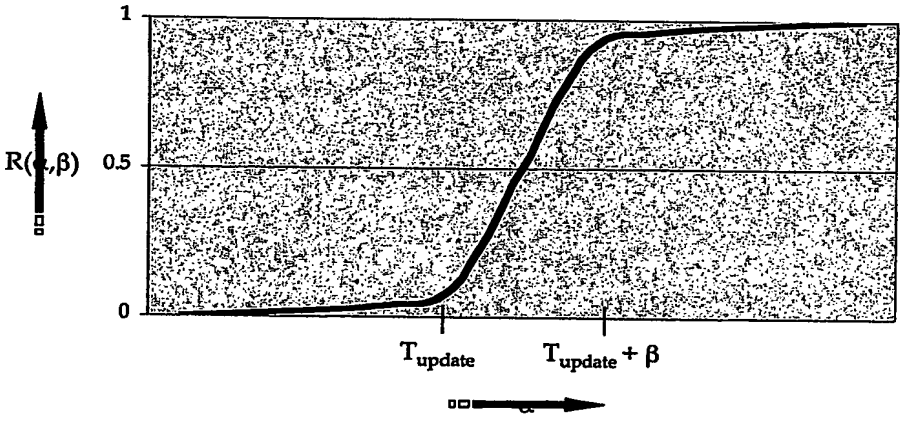


Figure 3: The Resistance Function, $R(\alpha,\beta)$, as a function of α for a given T_{update} and β

Figure 3 shows:

- If a system is exposed to a virus before the AV vendor has typically had a chance to update their signature file, i.e. $\alpha < T_{update}$, the ability of the AV product to resist the virus is essentially 0.0;
- If a system is exposed to a virus well after the AV vendor has updated their signature file and the user has downloaded and applied the up-

date, i.e. $\alpha >> T_{update} + \beta$, the ability of the AV product to resist the virus is essentially 1.0;

- In the transition region, where α lies between T_{update} and $T_{update} + \beta$, the value of $R(\alpha,\beta)$ rises as α increases, reflecting the increasing probability during traversal of this transition region of the virus' signature being found in the user's current AV signature file. This probability increases steadily as the virus ages under the assumption that there is no synchronisation between the AV vendor posting a virus signature update and the user checking to see if there is a new update available to be downloaded. (We are also, for simplicity in this example, assuming that β is large compared to the variance in T_{update} so the shape of $R(\alpha,\beta)$ across the transition is dominated by β and not by the variance in T_{update}.)

The probability of virus infection of the desktop is determined from $R(\alpha,\beta)$, the resistance function, and the virus threat number density $n(\alpha)$. That number density could, for e-mail-borne viruses, be measured by managed e-mail service providers. They could record over a period of time, say a month, the hourly-averaged proportion of e-mail messages that carry a virus as a function of that virus' age.

At any time there might be, say, two hundred active viruses in the wild, and for a highly active virus at its peak of activity, say seven days after its release, the virus might be present in, say, one per 250 e-mails. A typical virus threat number density might then look like the curve shown in Figure 4.

The probability, per Internet e-mail message received, of the desktop getting a virus infection is then the integral over all α of the product of $n(\alpha)$ and $P(\alpha,\beta)$ [which equals 1 - $R(\alpha,\beta)$]. This result will show how the probability of virus infection varies with the update period β and the volume of Internet e-mail messages handled. A user can then decide what probability of infection they are willing to tolerate and determine, given the volume of e-mail messages they receive on a typical day, how frequently they need to update their AV signature files in order to achieve that desired level of protection.

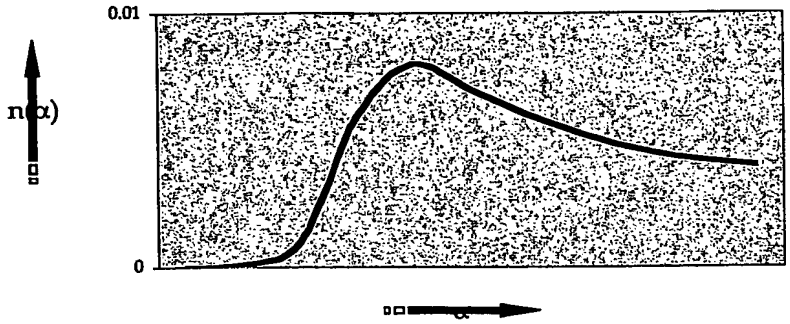


Figure 4: A possible number density for the virus threat, n(α), as a function of α

This example is a simple illustrative example based on a familiar interaction where it has been implicitly assumed that the infection of the desktop does not contribute to the subsequent number density of the virus threat, and that the rate of change of the virus threat over time is very small compared to the time taken for an infected host to expose a neighbouring host. This is clearly a suitable approximation for the Internet virus threat for which the magnitude of the threat at any time is achieved through the aggregated effect of many thousands of on-line infected hosts around the world. It is clearly not a valid assumption for the internal virus threat. This is covered in the next example.

(2) EXTENDING THE ABOVE EXAMPLE TO TWO THREAT ENVIRONMENTS

Consider now a second example which incorporates the internal virus threat. Consider a large network of desktops on a LAN where there is an AV product on the Internet mail gateway, protecting the user community from Internet-originated virus exposure, but not on the mail server handling internal e-mails. Then, for any desktop on the LAN, the probability per unit time of infection from a virus is the sum of the probability per unit time of infection from the Internet threat environment and the probability per unit time of infection from the internal threat environment. Once one of the desktops on the LAN becomes infected from the Internet, that desktop then contributes to the internal virus threat for all the other desktops on the LAN. The probability per unit time of infection for each of the other LAN-attached desktops starts to rise over time due to the growing exposure to the threat through the internal threat environment.

$$L(\beta,t) = \int n_{external}(\alpha).P_{external}(\alpha,\beta).\partial\alpha + \int n_{internal}(\alpha,t).P_{internal}(\alpha,\beta).\partial\alpha$$

In this situation, the evolution with time of the internal virus threat number density, $n_{internal}(\alpha,t)$, needs to be calculated. For this modelling, stochastic methods will be needed incorporating a function that describes the way that exposure propagates out through the infected host's local environment. The analysis shows how, in the absence of any Interdictive measures resisting the virus threat in the internal threat environment, i.e. for $P_{internal}(\alpha,\beta) \equiv 1.0$, each desktop's probability of virus infection grows as the number of desktops on the LAN grows.

This type of analysis would allow the security designer to model the affect of requiring each desktop to have a desktop AV product installed in addition to having the AV product reside on the Internet-facing mail server. This would allow the security designer to do a trade-off between his two options, installing the additional desktop AV products or just increasing the frequency of signature file checks (i.e. of improving β) for the mail gateway AV product.

(3) EXTENDING THE ANALYSIS TO MORE COMPLEX SCENARIOS

Multiple security measures, each contributing resistance to a given threat, can be included within the analysis by incorporating the resistance functions for each countermeasure. For example, if we take the first of the above virus examples and convert it into a worm example, a worm can be blocked in either of two ways: by its signature being in the AV signature files or by software patching to remove the software vulnerability the worm exploits. The two Interdictive measures can be treated as working in series, i.e. the threat has to dominate both the AV countermeasure and the patching countermeasure to be successful.

The threat distribution engaging the AV countermeasure is denoted by $n(\alpha)$. The distribution that remains after the interaction with the AV countermeasure is given by $L_{anti\text{-}virus}(\beta)$ where

$$L_{anti\text{-}virus}(\beta) = \int n(\alpha).P_{anti\text{-}virus}(\alpha,\beta).\partial\alpha.$$

This can be treated as an intermediate threat distribution which is then the distribution that engages the patching countermeasure. The interaction between the intermediate threat and the patching countermeasure is described by

$$L_{patching}(\beta) = \int L_{anti\text{-}virus}(\beta').P_{patching}(\beta',\beta).\partial\beta'$$

$$= \int n(\alpha).P_{anti\text{-}virus}(\alpha,\beta').P_{patching}(\beta',\beta).\partial\alpha.\partial\beta'.$$

The overall probability of success is determined by the product of the two constituent probabilities, and if the threat is defeated by either one of the two countermeasures, the overall probability is equal to zero.

The AV resistance function will again be as described above. The patching resistance function will be constructed through an analysis to determine which is the characteristic that determines the β variable for patching. With a small amount of reflection it can be seen that, for patching, the β variable is related to the period of time between release of the patch and the patch's installation. Just as there was a T_{update} value for the release of AV signature updates, there will be an equivalent T_{update} value for the release of a patch. The timescales are somewhat different, though, and the two protective strategies (an AV product and patching) are modelled as independent mechanisms:

- T_{patch_update} is typically several tens of days whereas T_{virus_update} is typically a few tens of hours;
- β_{patch} is typically measured in weeks whereas β_{virus} is typically measured in hours or days;
- The time from notification of a vulnerability to release of a worm exploiting that vulnerability is typically measured in months whereas the time from release of a virus in the wild to its peak of activity is typically measured in days.

This example shows how to model the threat environment reaching an internal information asset which is protected behind a number of Interdictive countermeasures. Consider the situation shown in Figure 5 where Server 1 is protected from the Internet by Firewall 1 alone and Server 2 is protected by both Firewall 1 and Firewall 2.

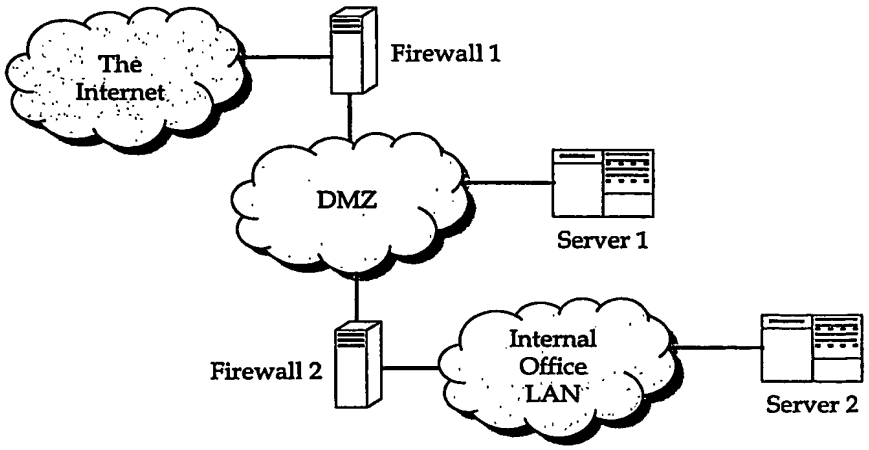


Figure 5: A two-layer firewall example

The threat distribution from the Internet threat environment is given by $n(\alpha)$. The Resistance function for Firewall 1 is given by $R_1(\alpha,\beta')$ and for Firewall 2 is given by $R_2(\beta',\beta)$. The threat distribution for hosts connected to the DMZ network is $n_1(\beta')$ where

$$n_1(\beta') = \int n(\alpha).P_1(\alpha,\beta').\partial\alpha.$$

This is the threat distribution reaching Server 1 and Firewall 2. The threat distribution reaching hosts connected to the Internal Office LAN, e.g. Server 2, is $n_2(\beta)$ where

$$n_2(\beta) = \int n_1(\beta').P_2(\beta',\beta).\partial\beta' = \int n(\alpha).P_1(\alpha,\beta').P_2(\beta',\beta).\partial\alpha.\partial\beta'.$$

This can be extended to cover any number of successive Interdictive countermeasures.

(4) PRELIMINARY RESULTS FOR OTHER INTERDICTIVE MEASURES

Determining the β variables and resistance functions for other Interdictive security measures follows similar reasoning. Using this approach has given us the following preliminary results:

- For AV countermeasures resisting the virus threat:
  - β is the promptness with which signature updates are applied;

- $R(\alpha,\beta)$ is as shown in the worked example above.

➢ For patching resisting the worm threat:

- β is the promptness with which patches are applied;
- $R(\alpha,\beta)$ is obtained from similar reasoning to that used for the AV case.

➢ For firewalls resisting the intrusion threat:

- β is determined by the frequency with which the firewall is vulnerability tested compared to the rate with which rule changes are made;
- $R(\alpha,\beta)$ is not completely unlike the shape of $R(\alpha,\beta)$ for the AV case, and it depends on:
  - ⇒ The technology used;
  - ⇒ The competence of the firewall installation and maintenance;
  - ⇒ The complexity and rate of change of the rule set;
  - ⇒ The frequency of vulnerability testing;
  - ⇒ The promptness with which vulnerability test findings are fixed.

➢ For User Security Education resisting the threat of security incidents caused by user errors and omissions:

- β is determined by how well, as a result of the security education, staff make basic security decisions for themselves;
- $R(\alpha,\beta)$ would initially be obtained by characterising security common sense (the desired result of security education) and then correlating comparative measurements of incidents experienced in different environments with that characterisation for each environment.

➢ For Corporate Security Culture resisting the threat of wilful misuse of privileges or failure to fulfil security responsibilities:

- β is determined by how well, as a result of strengthening the Security Culture, users voluntarily constrain their behaviour to conform to approved standards;
- $R(\alpha,\beta)$ would initially be obtained by characterising the psychological contract between employer and employee (a key part of how a corporate culture influences staff attitudes and behaviours) and then correlating comparative measurements of incidents experienced in different environments with that characterisation for each environment.

In some cases, and the two deterrent countermeasures mentioned above (User Security Education and Corporate Security Culture) are examples, it might not be immediately clear how to determine the resistance function $R(\alpha,\beta)$ from the type of analysis used with the AV countermeasure. In such cases, the security engineer will need to characterise the desired effect of the countermeasure and then correlate comparative measurements of incidents experienced in different environments with that characterisation for each environment in order to identify which countermeasure characteristics are most closely correlated with the idea of resistive strength. This approach is similar to benchmarking. With practice and experience, security designers and engineers will converge on a preferred way to characterise such countermeasures and agree a preferred variable for representing each β.

This analysis can be extended as indicated in the above examples to accommodate a wide variety of countermeasures, permitting the stochastic modelling analysis technique to be applied to a corporate IT infrastructure protected by a programme of security measures including technical and management measures, where host computers might be exposed to a combination of threat environments, where threats are resisted by a combination of security measures, and where the threat number density varies with time. Some of the countermeasure resistance functions might initially be known more precisely than others. In such situation, some of the resistance functions might need to be approximated until, with time and the analysis of gathered data, they can be estimated more accurately and precisely and then modelled analytically.

III. MODELLING THE SEVERITY OF SECURITY INCIDENTS

Referring back to our classification of security measures, the interactions which determine the severity of a process disruption caused by a successful attack are the interactions between the threat and the Mitigative countermeasures. Before we describe how these interactions are modelled, we shall

explain precisely what we mean by the term "severity" for a process disruption.

a) Definition of the terms Severity and Impact

When a security incident occurs, it has an impact on the business. That impact is brought about by the incident causing a disruption of some form to an important business process. That disruption might be to stop the continued performance of that process, perhaps by causing a system outage, or it might be to undermine the effectiveness of that process by, perhaps, causing the unauthorised disclosure of sensitive information where the process design has relied upon that sensitive information remaining confidential. Hence, incidents cause business impacts by triggering disruptions, partial or complete, of business processes involved in creating value for the company.

When a security incident causes such a process disruption, the magnitude of the business impact will depend upon the criticality of the process within the business operations as well as on the magnitude of the disruption created. For example, a two hour outage of a bank's trading floor will doubtless have a larger impact on the bank's business than a two-hour outage of one of its rural branches. The impact of a moderately severe unauthorised disclosure of sensitive information will be different from the impact of a moderately severe unauthorised corruption of critical data, which will in turn be different from the impact of a highly severe unauthorised corruption of critical data.

To accommodate these differences, we differentiate between the severity of a disruption and the impact that the disruption causes. For example, the severity of a power outage would be the duration of the outage (more particularly, the process time lost before the process is returned to operation). The resultant impact on the business of an outage of a given severity will depend on the criticality of the process affected and on other considerations such as whether the outage occurred during the working day or out of hours, whether it occurred just before a critical deadline, and so forth.

For each type of process disruption (we will discuss the different types of process disruption when discussing Step 1 later in this paper), we can define a severity variable *s*. A business analyst would be able to estimate the potential magnitude of the business impact arising from that disruption for each value of the severity variable. This could be done through a Business Impact Assessment or other similar process. The result would be charted as an impact function, *I(s)*, defined across the severity range. *I(s)* would normally be a rising function of *s*, and it might, perhaps, be similar to the example shown in Figure 6.

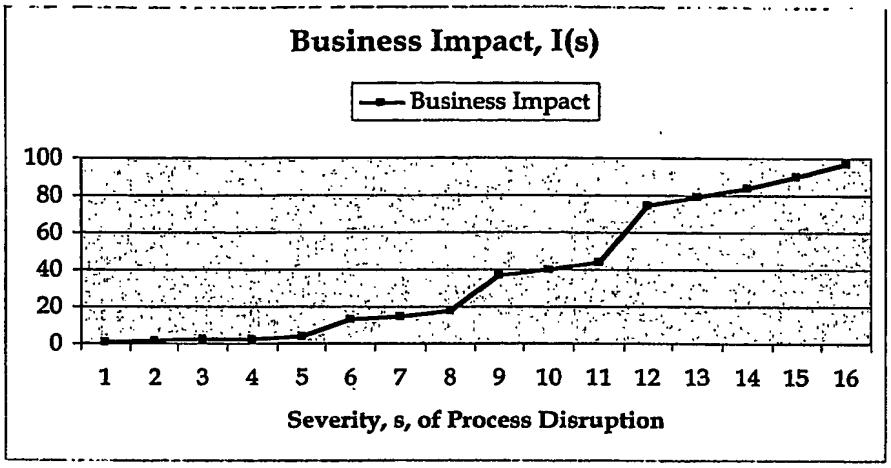


Figure 6: Business impact as a function of the severity of the process disruption

b) How Mitigative Security Measures Work

Each process disruption could be triggered by any of a variety of security incidents. For example, a process outage could be caused by, amongst other events, a crash of the system server, a power failure at the hosting data centre, a network failure between the server and the users. Each security incident is an instance of a particular attack. Whereas an incident, as a specific instance of an attack, gives rise to a disruption at a particular severity which could be measured for that incident, an attack could occur across a range of severities. For example, to say that the attack "network failure leading to SAP unavailability" could occur with a range of severities from *s1* to *s2* would mean that an instance of that attack would be a security incident causing an outage that lasted for at least a duration *s1* and for no more than a duration *s2*.

Each attack has a likelihood of causing a disruption at each of the possible severity values given by the (probably unknown) distribution *L(s)* (where we are now, for this discussion, suppressing the other variables not relevant to this part of the discussion, such as β). Figure 7 shows schematically three such attacks superimposed on the chart from Figure 6. Each attack causes disruptions across a range of severities and has a probability of causing a disruption at each severity value shown schematically in the form of a triangle. The width of the base of the triangle gives the range of severities which the attack can be expected to cause, and the height at any point on the triangle represents the likelihood of an instance of that attack causing a disruption at that severity.

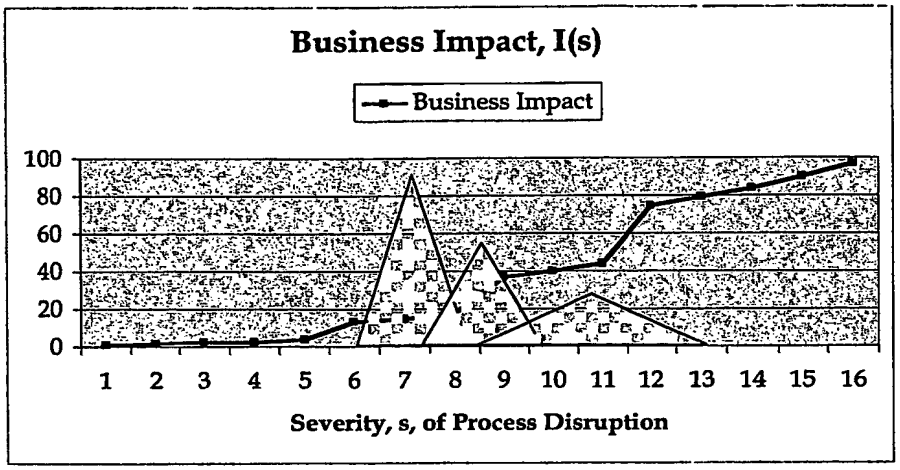


Figure 7: Three attacks triggering a process disruption

Now consider the effects of Mitigative security measures. Mitigative security measures reduce the severity of the process disruption from that which it would otherwise have been. For example, developing a good incident response capability allows a company to engage a team of trained and experienced personnel equipped with the right tools to try to reduce the severity of the disruption and thereby to reduce the impact of that disruption on the business. If the disruption is a process outage, they will work to get the process back on line as soon as possible. If the disruption is a fraud, they will work to recover as much as they can of the misapplied funds. If the disruption is an unauthorised disclosure, they will attempt to assess the extent of the disclosure and exert what influence they can to prevent the confidential information from getting into the hands of others that might use it to cause greater harm the company's interests.

The effect of Mitigative measures is to reduce the severity of the process disruption from that which it would otherwise have been. In terms of Figure 7, the effect of Mitigative measures is to move the attack triangles to the left. Mitigative measures do not reduce the probability of an incident occurring, so they do not reduce the area of each triangle, though in reducing the expected severity of each induced disruption they might reduce the range of severities over which disruptions would occur. In terms of Figure 7, this would appear as a reduction in the width of the base of the triangle as well as a shifting of the triangle to the left.

For example, a company might have a particular type of server in its data centre, and each time one of those servers fails it takes the company's operations staff between four and six hours to get the server back up. The company

might decide to take measures to reduce that time, with the result that afterwards each time one of those servers fails it takes the company's operations staff between two and three hours to get the server back up. Each disruption that would have occurred with a severity in the range of four to six hours will have been mapped to a disruption with a severity within the range two to three hours.

Based on this description, we model Mitigative measures as mapping functions, mapping unmitigated disruptions of severity γ to mitigated disruptions of severity δ. Mitigative measures are described by a Mitigation function $M(\gamma,\delta)$ where $M(\gamma,\delta)$ describes how disruptions that would have occurred with severity γ are mitigated and become disruptions occurring with severity δ. Note that the application or effect of any given Mitigative measure might vary with the type of attack being analysed.

The profile density of unmitigated disruptions is given by $L(\gamma)$ (again, suppressing those variables not relevant to this part of the discussion) where

$$\int L(\gamma).\partial\gamma = 1,$$

i.e. all incidents cause a disruption of some severity.

Mitigation does not change the number of disruptions occurring but transforms the severity profile from $L(\gamma)$ to $E(\delta)$, where

$$E(\delta) = \int L(\gamma).M(\gamma,\delta).\partial\gamma$$

and

$$\int E(\delta).\partial\delta = 1$$

The unmitigated, or baseline, severity of a disruption is determined by the innate ability of the organisation to recover from the attack in the absence of having any special measures in place. For example, if a critical process or service were to fail, staff would be able to recover it even if they had received no special training, guides or aids and had to make up the recovery process from scratch. It would just take unnecessarily long for them to recover the process, and the recovery would probably not be done in a well controlled and reliable way. The benefit of mitigative controls is that they reduce the baseline severity of the process disruption and thereby reduce the impact of the disruption.

c) Calculating the Mitigative Variables and Functions

As with the discussion of Interdictive measures, for this modelling technique to have value, we need to have confidence that we will be able to determine the variables γ and δ and the functions $L(\gamma)$, $E(\delta)$ and $M(\gamma,\delta)$ for any threat and countermeasure of interest.

γ, δ and the functions $L(\gamma)$, $E(\delta)$ and $M(\gamma,\delta)$ are obtained from an assessment of the way that the company would respond to a process disruption with and without the Mitigative measures in place. These will necessarily, in many cases, be imprecise estimation of response. However, they can be often be estimated with sufficient precision to be of good use within the Security Engineering analysis.

The first step is to estimate the baseline severity of incidents, that is the expected severity of different types of standard incident in the absence of any significant Mitigative measures being in place. It might well be the case that a company can assess baseline severity by looking back to one or a small number of examples of incidents it experienced before it put any focused incident response of other Mitigative measures in place. It could qualify its own experiences with data gathered from and shared by other organisations of a generally similar size. A number of membership bodies try to collect this type of data about the real-life experiences of their members, and though the data might be incomplete or patchy, it can serve to give a sufficient indication of the expected severity of standard types of incident when no significant Mitigative measures have been put in place.

It would not require a large number of data points to be gathered before it would be possible to make a sufficient estimation of baseline severities. In the absence of any contra-indications, severities could be describing in the form of a normal distribution around a mean value with an estimated variance given by the spread of the gathered data points. As with Interdictive measures, for a given standard security incident and system these assessments need be determined only once and agreed by consensus amongst security practitioners. With the passage of time and the gathering of increasing amounts of standardised and reliable data, the estimations of baseline severities would only improve.

A company can form its assessments of the expected severity of incidents after specific Mitigative measures have been put in place again by looking to its own data, the data of others, or its best estimations of what a planned Mitigative measure might deliver. It can be seen that this is not an inappropriate assessment to make if we consider, for example, the justification for implementing a Incident Response capability.

Incident response is primarily about having preparations and arrangements in place so that, in the event of a serious incident, an alarm can be raised promptly, people can deployed rapidly, and the incident dealt with by people with the relevant skill, experience and equipment. The objective is to achieve as promptly as possible a situation where the attack is causing no further damage (i.e. its severity is capped) and to return the process or service to full undisrupted operation as quickly and fully as possible.

As with any other type of fire-fighting service, it is natural to measure the strength of an incident response service in terms of its ability to deliver a defined level of service in defined situations, e.g.:

- To have the incident response team deployed within WW minutes of the alarm being raised;
- In XX% of cases to have the disruption's severity be capped within YY hours of the alarm being raised;
- In 100% of cases to have the severity capped within ZZ hours of the alarm being raised.

WW, XX, YY and ZZ then characterise the strength of the incident response capability. As before, in the absence of any contra-indications, severities could be describing in the form of a normal distribution around a mean value with an estimated variance given by the spread of the service data points. $M(\gamma,\delta)$ is then a simple mapping of one normal distribution, the unmitigated severity distribution characterised by a severity γ, to another normal distribution, the mitigated severity distribution characterised by a severity δ, with no loss in the total number of incidents.

Determining the γ and δ variables and $M(\gamma,\delta)$ functions for other Mitigative security measures follows similar reasoning.

IV. Modelling The Impact of Security Incidents

a) How Alleviative Security Measures Work

We have seen, through discussion and with the use of Figure 6 and Figure 7, how to envisage security incidents causing process disruptions of measurable severity. We were careful to distinguish between our use of the terms severity and impact, differentiating between severity as the measure of the intrinsic magnitude of a disruption and impact as the measure of the burden or business pain caused as a result of the disruption. Mitigative measures moderate the severity of a process disruption from that which it would otherwise have

been. The third group of security measures moderate the business impact of a process disruption for any given value of the severity variable. These we called the Alleviative security measures.

We shall follow a similar reasoning to that employed in the preceding section to describe the modelling of Alleviative measures.

When a security incident occurs and causes a process disruption, not only does the induced disruption have a given measurable severity but it has a given measurable impact. The magnitude of that business impact will depend upon the criticality of the process within the business operations and also on other considerations such as whether the outage occurred during the working day or out of hours, whether it occurred just before a critical deadline, and so forth. Hence, just as the severity of an individual disruption can vary within a severity range, so the impact can vary within an impact range.

When a business analyst assesses the impact of a disruption for different severities, as shown by the *I(s)* curve in Figure 6, they would normally estimate the largest impact that might reasonably be expected for that severity of disruption. However, the actual impact caused by any individual disruption will vary within a range from that level down. They will vary in a way that cannot be determined in advance because it cannot be known in advance exactly what the time and circumstances will be when the disruption occurs.

Alleviative measures reduce the impact of disruptions from that which they might otherwise have been, without having any effect upon the likelihood or severity of the disruption. For example, fallback systems allow mission-critical processes to continue, reducing the impact from the primary source of loss. Insurance transfers some of the cost of the impact onto the books of the insurer. In terms of the type of chart shown in Figure 7, Alleviative measures reduce the impact curve, as shown by the lower pink curve in Figure 8.

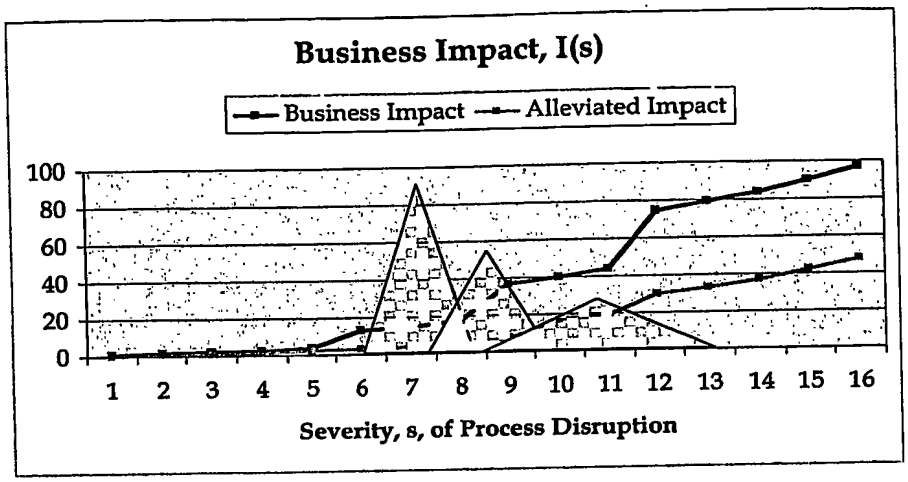


Figure 8: An example of the effect of Alleviative security measures

Just as we modelled Mitigative measures as mapping functions, mapping the baseline severity of disruptions, γ, to a lower mitigated severity, δ, we shall model Alleviative measures as mapping functions, mapping the baseline impact of disruptions, ε, to a lower alleviated impact, φ. Alleviative measures are described by an Alleviation function $A(\delta,\varepsilon,\varphi)$ where $A(\delta,\varepsilon,\varphi)$ describes how disruptions that would have occurred with severity δ and impact ε are alleviated and become disruptions occurring with severity δ and impact φ. Note that Alleviative security measures do not reduce the severity of the disruption, just as they do not reduce the likelihood of the disruption, though the application or effect of any given Alleviative measure might well vary with severity. Hence, the Alleviation function is $A(\delta,\varepsilon,\varphi)$ rather than more simply $A(\varepsilon,\varphi)$.

The profile density of mitigated but unalleviated incidents is given by $L(\delta,\varepsilon)$ (again, suppressing those variables not relevant to this part of the discussion) where

$$\int L(\delta,\varepsilon).\partial\varepsilon = 1,$$

i.e. all disruptions have some impact.

Alleviation does not change the number of disruptions that occur but transforms the impact profile from $L(\delta,\varepsilon)$ to $E(\delta,\varphi)$, where

$$E(\delta,\varphi) = \int L(\delta,\varepsilon).A(\delta,\varepsilon,\varphi).\partial\varepsilon$$

and

$$\int E(\delta,\varphi).\partial\varphi = 1$$

b) Calculating the Alleviative Variables and Functions

As with the discussion of Interdictive and Mitigative measures, for this modelling technique to have value, we need to have confidence that we will be able to determine the variables ε and φ and the functions $L(\delta,\varepsilon)$, $E(\delta,\varphi)$ and $A(\delta,\varepsilon,\varphi)$ for any process disruption and countermeasure of interest.

ε and φ and the functions $L(\delta,\varepsilon)$, $E(\delta,\varphi)$ and $A(\delta,\varepsilon,\varphi)$ are obtained from an assessment of the impact of a process disruption with and without the Alleviative measures in place. These will necessarily, in many cases, be imprecise estimation of response. However, they can be often be estimated with sufficient precision to be of good use within the Security Engineering analysis.

The first step is to estimate the baseline impact of a given disruption, that is the expected largest impact that might reasonably be expected for that disruption, as a function of the severity of that disruption and in the absence of any significant Alleviative measures being in place. This is an assessment that the relevant business unit should be able to perform based upon its understanding of its business and the interdependencies of various important value-generating processes within its business.

The second step is to estimate the impact of a given disruption after the selected Alleviative measures have been applied, that is the expected largest impact that might reasonably be expected for that disruption, as a function of severity. This, too, is an assessment that the relevant business unit should be able to perform. For example, if the Alleviative measure is a fallback process, this assessment might require a similar type of analysis to that performed for the baseline impact. Alternatively, if it is insurance, it will be determined simply from inspection of the insurance cover obtained.

The third step is to consider what the distributions of impacts might be at each severity value for each of the baseline and alleviated cases. This will depend on the nature of the disruption. For example, if the disruption is a fraud within a payment system, and the maximum impact that might be incurred is, say, £10 million (because the system is not used to transfer larger sums than

that), then the maximum impact at 100% severity would be £10 million. The maximum impact at each lesser severity would be £10 million scaled down linearly according to the value of the severity. The question then asked is what might the expected distribution of actual impacts be below that maximum at each level of severity. The business unit might decide that the fraudster would always opt for the maximum size of fraud they could commit, in which case the distribution collapses down to a single value, the maximum value at each severity. Alternatively, the business unit might decide that fraudsters would choose the size of the fraud randomly and model $L(\delta,\varepsilon)$ as a normal distribution with the maximum reasonably likely value of $L(\delta,\varepsilon)$ being $I(\delta)$.

Again, the business unit might look to data gathered from and shared by other organisations or membership bodies to help support its assessment.

Determining the variables ε and φ and the functions $L(\delta,\varepsilon)$, $E(\delta,\varphi)$ and $A(\delta,\varepsilon,\varphi)$ for other process disruptions and countermeasure of interest follows similar reasoning.

V. THE THREAT PROFILE FUNCTIONS

In the above discussions, we have referred to a number of functions which describe the number density of the threats and the baseline severity and impacts of disruptions. In each part of the discussion, we have, for the sake of clarity, not shown those of the functions' variables that have not been germane to that part of the discussion. It is appropriate, now that the discussion has covered how to model the interaction of threats with each of the three classes of security measure, to show the threat number density functions in full.

In Section 3.B.II we denoted the threat number density function as $n(\alpha)$. In Section 3.B.III we denoted the likelihood of each process disruption occurring at each severity value as $L(\gamma)$. In Section 3.B.IV we denoted the likelihood of each process disruption occurring at each impact value as $L(\delta,\varepsilon)$. In practice, $L(\gamma)$ and $L(\delta,\varepsilon)$ are each manifestations of the full threat number density function, which should be written in full as $n(t,\alpha,\gamma,\varepsilon)$. We refer to this function, $n(t,\alpha,\gamma,\varepsilon)$, as the threat profile function or number density.

The likelihood per unit time of an incident occurring due to a threat described by $n(t,\alpha,\gamma,\varepsilon)$ resisted by a security measure of strength β is given by the function

$$L(t,\beta,\gamma,\varepsilon) = \int n(t,\alpha,\gamma,\varepsilon).P(\alpha,\beta).\partial\alpha.$$

We have defined $L(t,\beta,\gamma,\varepsilon)$ such that

$$\int L(t,\beta,\gamma,\varepsilon).\partial\gamma = 1 \text{ and } \int L(t,\beta,\gamma,\varepsilon).\partial\varepsilon = 1.$$

Hence, what we described as $L(\beta)$ in Section 3.B.II can be expressed as

$$L(\beta) = L(t,\beta) = \int L(t,\beta,\gamma,\varepsilon).\partial\gamma.\partial\varepsilon = \int n(t,\alpha,\gamma,\varepsilon).P(\alpha,\beta).\partial\alpha.\partial\gamma.\partial\varepsilon.$$

C. THE FOUR SECURITY ENGINEERING STEPS

We shall now describe each of the four Security Engineering steps in turn (refer back to Figure 2 for an overview of the four steps).

I. STEP 1

Step 1 is the process of defining the security targets that the security designer will aim to satisfy.

Conventionally, the security need, the statement of security direction given to the security designer, is expressed either in terms of a security classification, which is often a High / Medium / Low classification for each of Data Confidentiality, Integrity and Availability resulting from a risk assessment, or in the form of a Value-at-Risk estimate resulting from an impact assessment. This provides the designer with a statement of the magnitude of the security need. However, it does not provide the designer with specific targets for the security outcomes the design needs to achieve.

Security Engineering expresses the security need in the form of security targets, with the intention that the security targets should be as objective and measurable as reasonably practicable.

Step 1 is performed in six stages as shown in Figure 9.

a) Stage 1.1

The process starts with the Business Process Owner (BPO) describing the nature of the system needing protection (e.g. supply-chain, transactional, SCADA, management information), the expected size of the system (e.g. expected number of users, transactions, data volumes, and so forth) and identifying the key business processes and their operational objectives. These operational objectives are the objectives that the business processes need to achieve if they are to be judged successful by the relevant business unit. The operational objectives might be expressed as essential functional objectives (e.g. ensuring payments are sent out before the daily cut-off time) or as non-functional objectives (e.g. privacy, reliability, ease of use, low cost of support and maintenance).

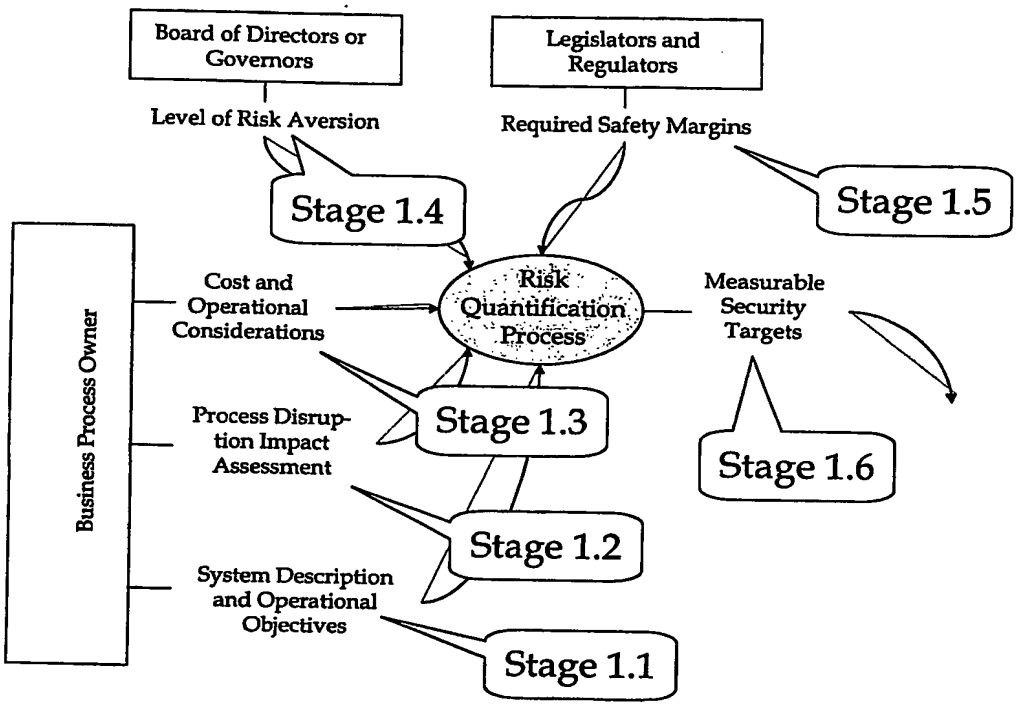


Figure 9: The six stages of Step 1

The BPO then identifies the ways that the system's main process can be made to fall short of, or fail to achieve, their operational objectives (e.g. process stoppage, inaccurate or unreliable data, fraud, critical functional or control failures). These will expose the disruptions that the business processes need to be protected against. To the degree that the business processes are implemented through, and reliant on, an IT system, these are the failure modes that the IT system needs to be protected against.

b) Stage 1.2

In Stage 1.2, the business' sensitivity to process disruptions is calculated. This will determine if the system is going to need lax or tight security targets.

For each disruption, the BPO assesses the potential magnitude of the business impact arising from that disruption occurring at each value of severity. This can be performed by a Business Impact Assessment, structured interviews, scenario exploration, etc. Importantly, the impact is to be assessed not just in the form of a single "worst case" impact value but as a variable parameter $I(s)$ defined across the severity range. $I(s)$ would normally be a rising function of s and needs to be estimated only as precisely as the business unit can readily

estimate it; it does not need to be estimated with high precision. *I(s)* is discussed in section 3.B.IV above.

As well as being characterised by the Impact function, *I(s)*, process disruptions are also characterised by their probability function, *L(s)*. *L(s)* is the aggregated result of the likelihood of each contributing attack and would normally be a generally falling function of *s*. Its actual value in any operational environment is not likely to be known unless it has been possible to collect a large amount of reliable and standardised data from relevant security incidents. *L(s)* is not measured or calculated at this stage of the process. It is used in Stage 1.6 as the basis on which the security targets for the designer are expressed. It is calculated in Step 4 from the calculated distributions $L_A(s)$ of each of the major contributing attacks A.

c) Stage 1.3

In Stage 1.3, the non-security considerations that will influence the details of the security design are specified. These will be used in Step 4 to guide the optimisation part of the design process.

It will be recognised in any realistic operational environment that the business unit will not want to obtain security protection at any price. Security measures tend to impact a number of other system requirements such as cost, performance, ease of use, and a business unit is not likely to accept a security design that gives rise to an unacceptable impairment of these other system requirements.

The BPO identifies the non-security constraints (e.g. cost, performance, reliability, ease of use) that will be considered in the optimisation of the security design. Each constraint can be expressed in whatever form the business process owner finds most appropriate, for example:

- The total cost of the security products cannot exceed £XX,000;
- If the throughput is likely to fall below YY transactions a second, refer back to the business process owner before proceeding;
- "Ease of use" is our top priority and the security measures must not significantly complicate or impair the system's usability;
- And so forth.

The more objective and measurable these constraints can be made, the easier it will be for the security designer to optimise the security design around them.

The difference between the operational constraints introduced in this stage and the operational objectives introduced in Stage 1.1 is this. The operational objectives introduced in Stage 1.1 are the capabilities of the business process that the security designer is charged directly to protect to the specified level. The operational constraints introduced here, in Stage 1.3, are not protected directly by the security measures but are required not to be impaired unduly as a consequence of the security measures. The boundary between what is normally directly protected (objectives) and what is normally indirectly protected (constraints) is a matter of local practice.

d) Stage 1.4

In Stage 1.4, the BPO assesses senior management's stated appetite for risk and takes management's guidance into consideration.

Before the BPO determines the security metrics and security targets for the security designer, he should review applicable internal governance mandates, security policies and standards for guidance on the level or nature of protection and assurance senior management would like to see practised. This will help the BPO to determine the priority that should be given to security against other information management concerns, and might dictate a baseline or minimum level of security protection to be sought.

Conventionally, governance mandates and security documentation express senior management's position as a mixture of subjective wishes and standards to be applied with some discretion. It is to be hoped that, as the Security Engineering techniques take hold, it will be possible for these internal guidance statements to become more objective and measurable.

e) Stage 1.5

In Stage 1.5, the BPO assesses any applicable regulation and legislation regarding the level or nature of protection mandated.

The BPO should review applicable external regulation and legislation, for example legislation relating to data protection, records management, sectoral regulations, and so forth, bearing in mind that Internet-accessible systems might need to comply with regulations from a number of jurisdictions. The guidance, mandates or standards contained in the regulations and legislation might dictate a baseline or minimum level of security protection to be sought.

Conventionally, regulation and legislation express security requirements in a wholly subjective and immeasurable way using phrases such as "appropriate technical measures". It is to be hoped that, as the Security Engineering

techniques take hold, it will be possible for these statements to become more objective and measurable.

f) Stage 1.6

In Stage 1.6, the BPO combines the inputs from the preceding stages and formulates the security targets that are to be achieved by the security design.

It will be recognised in any realistic operational environment that absolute security is not practicably achievable and that near-absolute security would normally create unacceptable cost and operational burdens. Hence, the BPO states the desired security targets in the form of limits on the amounts of insecurity that the business unit is prepared to tolerate. This is very similar to the way that service availability targets are usually expressed. For example, if a BPO requests "5 9's availability" that says that the business unit is prepared to tolerate 0.005% unavailability but not more.

The BPO has first to identify the metrics he will use for measuring the level of insecurity achieved. He will then define the target to be achieved in each metric.

The metrics can be expressed in a variety of possible forms (c.f., for example, NIST Special Publication 800-55 "*Security Metrics Guide for IT Systems*") reflecting the measurement capabilities of the organisation. Metrics can be measures of:

- The impact of security disruptions (e.g. the aggregated impact of all outages within a rolling time window);
- The level of process disruptions (e.g. the number and severity of outages per year);
- The level of particular attacks (e.g. the number of intrusions per year);
- The effectiveness of specific types of security measure (e.g. the proportion of staff passing an in-house "security knowledge and awareness" test each year);
- The implementation of specific types of security measure (e.g. the number of staff taken through an in-house security training course each year).

Each organisation is likely to employ a variety of different types of metric according to the system or processes being protected. In general, the aim should be to prefer metrics which are closer to the business impact (i.e. those

types higher in the above list) and which can support objective and measurable targets. When employing metrics which are not directly associated with the business impact, for example metrics which are expressed in terms of attacks or security measures, it is important to identify meaningful metrics that will contribute to reducing the business impact of insecurity.

The security targets set by the BPO are measurable targets for each of the identified security metrics. They can be expressed as either a single threshold target or as a small group of thresholds in each metric. For example, the targets could be such as:

- No more than £100,000 in total fraud losses per year:
- No more than 5p per electronic shopping transaction in fraud and deception losses;
- No more than 2 hours of aggregate outage over a rolling year;
- No more than one outage per month ≤ 15 minutes in duration and no more than one outage per year ≥ 15 minutes duration;
- No more than one virus infection per workstation per month;
- 80% of critical server security patches to be applied within three days and 99% to be applied within 14 days.

As the security targets are to form the objectives for the security designer, it would make sense for the BPO to formulate them in consultation with the security designer in order to ensure they are properly understood and likely to be achievable.

II. STEP 2

Step 2 is the process of analysing and measuring the threat environments.

The goal in Step 2 is to identify and then quantify the threats working to induce disruptions to the protected business processes. Conventionally, security practitioners try to identify all the attacks they believe to be important and then to estimate the probability of each attack occurring. This process is plagued by very poor quality data and does not support forecasting. In Step 2, the Security Engineering approach is to quantify the threats within each threat environment rather than to quantify the incidents seen. The influence of the quantified threats on the resultant incidents will be obtained by modelling in Step 4.

Step 2 is performed in four stages as shown in Figure 10.

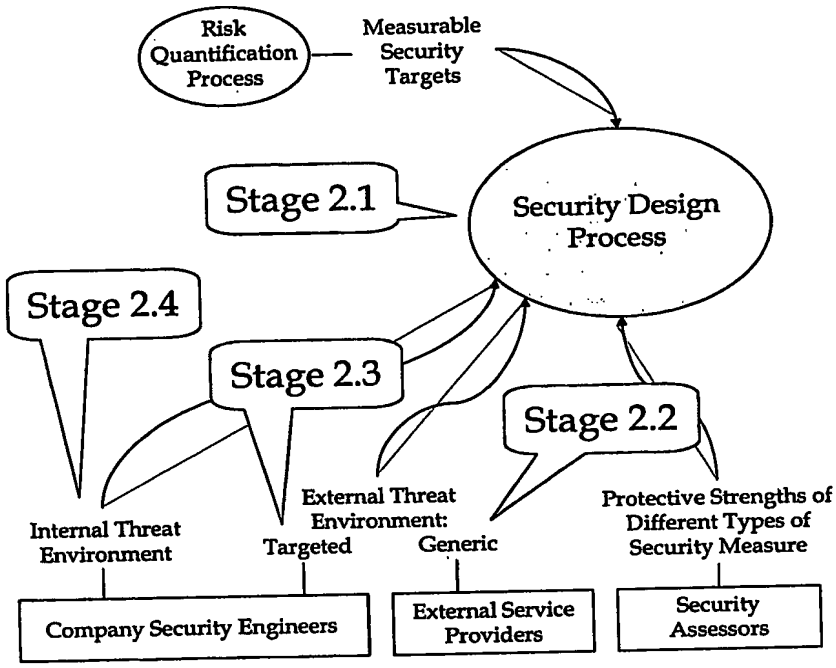


Figure 10: The four stages of Step 2

a) Stage 2.1

In Stage 2.1, the security designer identifies the threats relevant to the business process disruptions that require protection, and then determines the parameters that will be used to quantify the threats.

In the first part of Stage 2.1, the security designer identifies the relevant threats. This is done by identifying the process disruptions of interest to the security targets, identifying the main attacks likely to trigger those process disruptions, and then identifying the threats and threat agents that lead to those attacks.

It should be evident which are the process disruptions of interest to the security targets. They are most likely to be explicit in the statement of security targets, and if not should be evident from the description of the business process given by the BPO in Stage 1.1.

The security designer is free to identify attacks using whichever method he believes to be most suitable and appropriate. However, it is of course important that the process should be as close to exhaustive as possible.

The conventional manual way to identify attacks is through brainstorming with one or more local experts. This is not exhaustive and cannot be expected to identify attacks other than those that the local experts are most familiar with at the time.

The conventional automated way to identify attacks is through the use of a software Risk Assessment tool containing an inventory of attacks. This is no more exhaustive than the inventory of attacks contained within the tool and it relies heavily on the tool having a sound classification system so that relevant attacks from within the inventory can be extracted reliably for each type of disruption. There is at the time of writing no established or proven standard for such an attack classification system.

Both of these approaches also suffer from their reliance on the familiarity of the local experts or tool maintainers with the types of disruption under consideration. This might not be a weakness when only familiar disruptions are under consideration, for example disruptions associated with attacks on the Confidentiality, Integrity or Availability of data. However, Security Engineering can be used to great effect in situations where the security designer might not be so familiar with the disruptions of concern, for example a disruption in the ease of use or reliability of the protected process. In these situations, common experience cannot be relied upon to lead to the exhaustive or near exhaustive identification of relevant attacks.

The recommended approach is to use a Taxonomy of Threats, Attacks and Incidents such as the taxonomy shown in Figure 11. This taxonomy has been designed to identify attacks brought about by human error, carelessness and negligence as well as by human will or malice, and attacks which are normally considered to be accidents, i.e. brought about by serendipity.

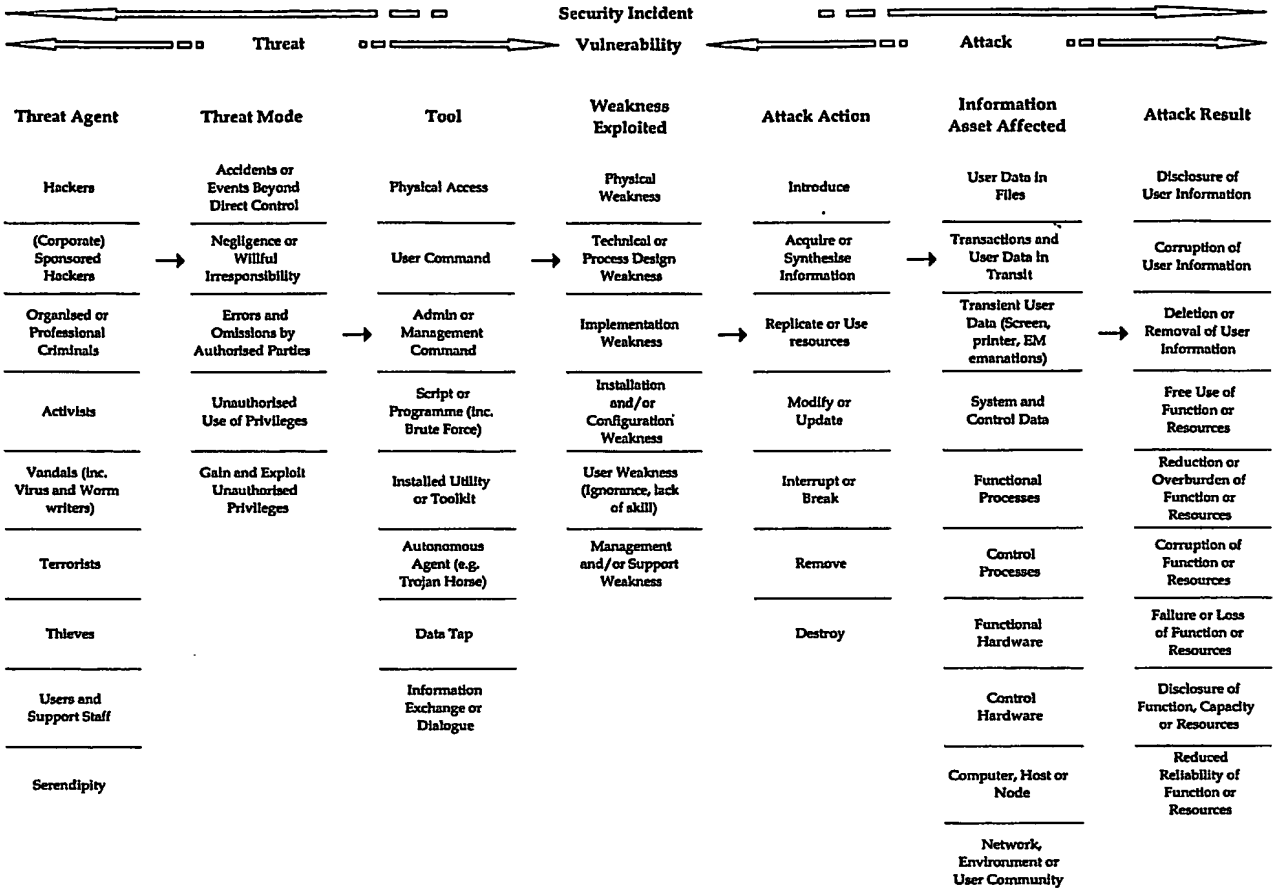


Figure 11: The Taxonomy of Threats, Attacks and Incidents

Through the use of this Taxonomy, or through alternative methods, the security designer identifies all the attacks that are applicable to the system or process requiring protection. In practice, the security designer should, at this stage, filter out those attacks which he is confident will always be either very unlikely or never lead to significant impacts. This narrows the attack inventory down to only those attacks which are significant and improves the tractability of the process.

The security designer then identifies the threats that can lead to those selected attacks. These are the threats which will be quantified in the rest of Step 2 in preparation for the modelling step, Step 4. If a taxonomy was used to identify relevant attacks, the threats will have been identified in the process of identifying attacks.

Having identified the relevant threats, the security designer now identifies the parameters to be used for quantifying the threats. These are the α, γ and ε variables as described above in Sections 3.B.II, 3.B.III and 3.B.IV. For a given threat, these variables need be determined only once for each process and security measure. In many cases, they might be independent of the specific process or security measure, which clearly simplifies the analysis and threat modelling. Once the variables have been agreed by consensus amongst security practitioners, they are set and do not need to determined afresh for each analysis. Hence, the security designer identifies the α, γ and ε variables from accepted practice, where that exists, or by analysis as discussed in Section 3.B.II onwards.

The rest of Step 2, Stages 2.2 onwards, concern themselves with the measurement of the number density of the threats for each threat environment. These are the threat profile $n(\alpha)$ functions discussed in Sections 3.B.II to 3.B.V. Each $n(\alpha)$ function describes the threat within each of the prevailing threat environments and will vary from threat to threat, from threat environment to threat environment, and perhaps from day to day or month to month. The $n(\alpha)$ functions are obtained by measurement and are needed for each threat and threat environment.

b) Stages 2.2 Onwards - The measurement of the $n(\alpha)$ functions

There is much work to be done in gathering threat measurement data in a form appropriate for Security Engineering modelling. The Information Security industry does not today (late 2003) have much data of use for these purposes. This is not because such data is inherently difficult to obtain. It is largely because the industry has up to now lacked agreement as to what analysis it would be useful to perform and, therefore, what data it would be useful to gather and how that data should be presented. Without a framework to guide efforts in this area, many of the organisations which are in a position to collect data have tended to collect that which is readily collectable without being guided by any insight as to how the data should be standardised, made reliable, labeled and presented for useful analysis. Security Engineering provides a clear framework for the gathering of standardised and reliable threat measurements.

What is sought is measurement of the threat number density functions for relevant threats and threat environments. This is most easily and accurately performed in situations where many organisations see the same level of threat for a given threat and threat environment. Hence, threat number densities can be measured most readily for generic external threats where

there is ample data available for collection and all users see essentially the same threat climate.

Measurements of generic external threats (primarily viruses, worms, and attacks against the Internet infrastructure itself) could be undertaken by various commercial or non-commercial organisations, for example:

- For the virus threat – Managed mail service providers (such as MessageLabs and others) in partnership with AV vendors (such as Symantec and others) or groups (such as the Wildlist organisations) would be well placed to measure this threat;
- For the worm threat – Managed security services providers (such as Symantec, Ubizen and others) in partnership with vulnerability analysis service providers (such as TruSecure and others) would be well placed to measure this threat;
- For threats against the Internet infrastructure itself – collaborative and research groupings (such as SANS, CAIDA, Carnegie Mellon CERT, and others) would be well placed to measure these threats.

One can imagine service providers producing a daily or weekly "weather report" for each of these types of threat, providing a regular update on the general threat climate, showing local variations from sector to sector, and providing a forecast for the coming weeks or months. With growing practice and experience, providers would be able to provide more accurate forecasts of how the threat climate should be expected to change as a result of significant external events such as major geo-political upheavals or technological advances.

Threat number densities would take more time to measure, and would most likely remain less accurate, for targeted external threats (primarily attacks against web servers and Denial of Service attacks). There is less data available for collection and different users will see a different threat climate, meaning that the available data would need to be more granularly presented. Measurements of targeted external threats could be undertaken by the same or similar organisations to those gathering generic external threat data. For example, managed security services providers (such as Symantec, Ubizen and others) would be well placed to measure this threat.

Measurements of the internal threat (primarily different types of inappropriate behaviour by staff) would, in almost all circumstances, need to be measured by the organisations themselves. The difficulty here is that each organisation, in measuring its own internal threat environment, would be

measuring the threat climate that it faced after the application of its internal security measures. Hence, different organisations would measure different threat number densities.

However, notwithstanding that, there are strong reasons to believe that what might be called the underlying threat number density, the threat number density that would be found if there were no relevant security measures in place, would be very similar between organisations of a similar size. All organisations within each geographical region recruit staff from essentially the same labour pool and their staff use essentially the same types of information technologies and products in their daily work. Other than variations caused by local security measures, staff can be expected to display essentially the same range of attitudes and behaviours whichever company within a given sector they might work for. This means that variations between companies would be largely due to differences in their security measures (education, culture, etc.) and, through the sharing of data across a range of companies, it would be possible to define a common baseline from which each organisation could measure the benefits of the security measures it applied. With the gathering of greater volumes of data over time, the baseline would be refined.

Measurements of the internal threat, accounting for four of the five threat modes shown in the Taxonomy of Threats, Attacks and Incidents (see Figure 11), could come from various internal sources. Data could be gathered from internal incident reporting schemes, HR records, help desks, and so forth.

It is to be hoped that, as the Security Engineering techniques take hold, consensus will arise regarding both the data required and appropriate standards for the gathering of that data. A thriving industry in gathering and sharing standardised and reliable data about a wide range of threats would then be able to emerge. It has been recognised since at least the mid 1990's that there is great value to be gained from the widespread sharing of good threat and incident data. Security Engineering, by providing a framework for data gathering and sharing, can help overcome the barriers that have so far prevented this goal from being fulfilled.

III. STEPS 3 AND 4

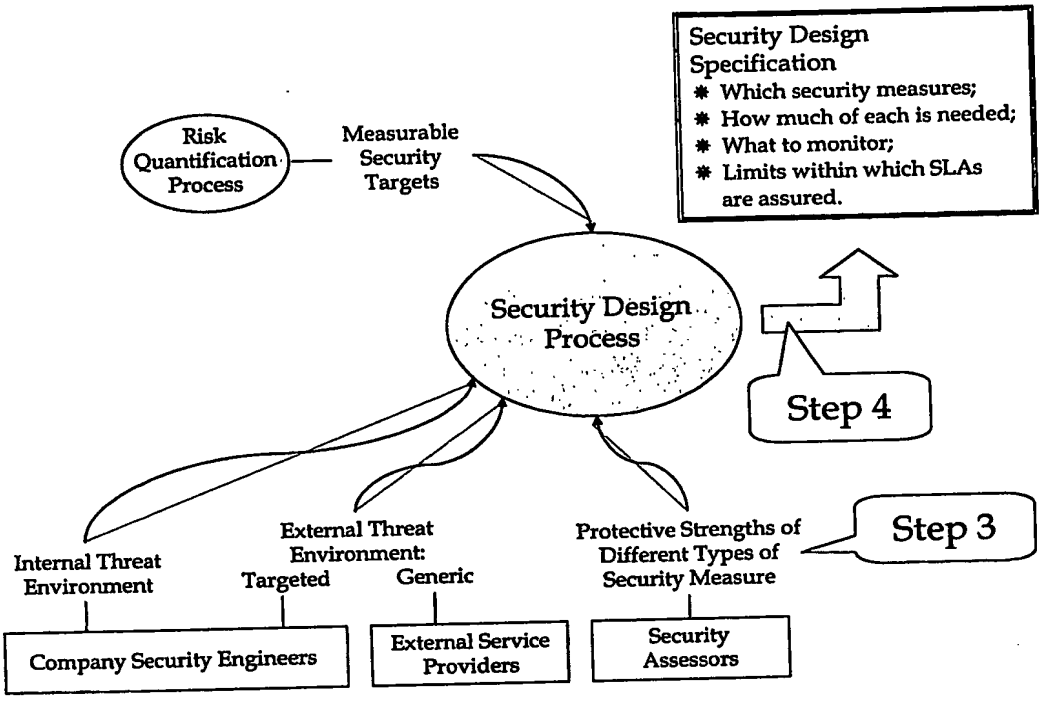


Figure 12: Steps 3 and 4

Step 2 focused on gathering all the information needed for modelling the threats. Step 3 focuses on gathering all the information needed for modelling the security measures. Step 4 is where the modelling is performed and security design optimised.

There is a variety of information that the security designer will need to compile in Step 3. It includes:

- The Resistance, Mitigation and Alleviation functions for each of the security measures that might be required. Where the security measures are widely used standard measures, such as Anti-Virus products, patching, firewalls, etc., these functions will be well known and any relevant data close to hand. When this data is first being compiled, it will come from a variety of external and internal independent trusted sources.

- Measurements of how well the company is able to implement each security measure. For example:

- Is the company able to update its AV signature files as often as needed or is there a limit to how often that can be performed?
- How quickly can critical security patches be applied to different technology platforms?
- How long does it take, typically, for the results of a firewall vulnerability test to be applied?
- How strong is the corporate security culture and how long will it take to effect a significant improvement in its strength given the current strength of the corporate culture generally? (The stronger the corporate culture, the easier it is for management to change aspects of that culture; the weaker the corporate culture, the harder it is to build up any aspect of that culture.)

➢ Measurements of the cost of each security measure (prices paid, effort expended, etc.)

➢ Measurements of the impact of each security measure on each of the operational constraints identified earlier in Stage 1.3.

In Step 4, the modelling is performed, the security measures selected, and the design optimised to satisfy the security targets whilst retaining the best fit to the operational constraints.

The objective of modelling is to forecast the level and nature of process disruptions induced by a quantified level of each threat when pitted against a specified amount of each security measure. The quantity of each security measure can then be increased or decreased until the forecast level of incidents satisfies the security targets set for the design.

For each security target, the security designer will need to review which process disruptions are relevant, which threats might induce those disruptions, and which security measures might be suitable for reducing the likelihood of successful attacks or the severity or impact of the disruptions. The security designer will also determine by inspection of the security targets what degree of latitude he has for satisfying the targets through varying the mix of different types of security measure.

The security designer will be guided in his selection of which security measures to work with by the body of local established practice built up over past experiences using the Security Engineering modelling techniques. Different organisations will have different abilities to apply and enforce different

security measures which will lead to differences in their preferred ways of achieving the security targets set. Different systems will also have different operational constraints which will also lead to differences in the ways to achieve the security targets set.

Starting with the measured threat profiles, the security designer models the evolution of threat profiles throughout the system and the disruptions forecast. Software tools will be needed for the numerical computations called for in the modelling. The security designer adjusts the security measures used until the forecast level of disruptions satisfies the security targets. The impact of the security design in terms of cost and operational impact is then calculated. The security design is varied to explore other ways of satisfying the security targets, and the option that fits best with the cost and operational constraints would normally be the preferred option.

As well as varying the design to fit with the cost and operational constraints, the security designer will also track the sensitivity of the results to variations in the input data. For example, the values used for the threat profiles will inevitably contain a degree of uncertainty. The designer will want to assess the affects of having a higher threat profile partly to allow for any inaccuracies in the threat profile used for modelling and partly to allow for any future growth or changes in the actual threat profile experienced. The designer will want to assess the sensitivity of the results to the level of security measure employed. The company might believe itself able to exercise security measures to a stated degree but might then find itself unable to achieve that level in full. Selecting a security design that is robust under such eventualities will be part of the skill of the security designer.

This sensitivity analysis will indicate to the security designer which are the key signs of security distress to be monitored during the operation of the protected system. The designer will advise on what monitoring should be undertaken, including what monitoring data to collect and how frequently it is to be assessed, so that any signs that the security targets might not be satisfied can be detected promptly and escalated to security operations management or business management for action. Management might decide to strengthen the security measures temporarily to ride a period of heightened threat (e.g. the detection of a new virulent worm in the wild or increased hacking activity against an Internet gateway) or to curtail the system's operation in some way in order to reduce the security stresses on the system.

The final security design, with its analysis of sensitivities, recommendations for monitoring, and fit to the cost and operational constraints is then put to the originating BPO for acceptance and approval. The security designer will advise the BPO on the operational limits within which the security design can

be expected to achieve the security targets set. This will allow the BPO, when the system is up and running, to monitor the operation of the system and to judge at what stage the level of use being made of the system might indicate the security design should be reviewed and revised.

--- o --- End of Paper --- o ---